

E.André, J. Müller, T.Rist



Deutsches
Forschungszentrum
für Künstliche
Intelligenz GmbH

**Research
Report**

RR-96-02

PPP-Persona: Ein objektorientierter Multimedia-Präsentationsagent

E.André, J. Müller, T.Rist

März 1996

**Deutsches Forschungszentrum für Künstliche Intelligenz
GmbH**

Postfach 20 80
67608 Kaiserslautern, FRG
Tel.: + 49 (631) 205-3211
Fax: + 49 (631) 205-3210

Stuhlsatzenhausweg 3
66123 Saarbrücken, FRG
Tel.: + 49 (681) 302-5252
Fax: + 49 (681) 302-5341

Deutsches Forschungszentrum für Künstliche Intelligenz

The German Research Center for Artificial Intelligence (Deutsches Forschungszentrum für Künstliche Intelligenz, DFKI) with sites in Kaiserslautern and Saarbrücken is a non-profit organization which was founded in 1988. The shareholder companies are Atlas Elektronik, Daimler-Benz, Fraunhofer Gesellschaft, GMD, IBM, Insiders, Mannesmann-Kienzle, Sema Group, Siemens and Siemens-Nixdorf. Research projects conducted at the DFKI are funded by the German Ministry of Education, Science, Research and Technology, by the shareholder companies, or by other industrial contracts.

The DFKI conducts application-oriented basic research in the field of artificial intelligence and other related subfields of computer science. The overall goal is to construct systems with technical knowledge and common sense which - by using AI methods - implement a problem solution for a selected application area. Currently, there are the following research areas at the DFKI:

- Intelligent Engineering Systems
- Intelligent User Interfaces
- Computer Linguistics
- Programming Systems
- Deduction and Multiagent Systems
- Document Analysis and Office Automation.

The DFKI strives at making its research results available to the scientific community. There exist many contacts to domestic and foreign research institutions, both in academy and industry. The DFKI hosts technology transfer workshops for shareholders and other interested groups in order to inform about the current state of research.

From its beginning, the DFKI has provided an attractive working environment for AI researchers from Germany and from all over the world. The goal is to have a staff of about 100 researchers at the end of the building-up phase.

Dr. Dr. D. Ruland
Director

PPP-Persona: Ein objektorientierter Multimedia-Präsentationsagent

E.André, J. Müller, T.Rist

DFKI-RR-96-02

A version of this paper has been published in:
W.Wahlster (Hrsg.). Fortschritte der objektorientierten Softwaretechnologien.
Congress VI. ONLINE '96.

This work has been supported by a grant from The Federal Ministry of Education, Science, Research and Technology (FKZ ITWM-9400).

© Deutsches Forschungszentrum für Künstliche Intelligenz 1996

This work may not be copied or reproduced in whole or part for any commercial purpose. Permission to copy in whole or part without payment of fee is granted for nonprofit educational and research purposes provided that all such whole or partial copies include the following: a notice that such copying is by permission of the Deutsche Forschungszentrum für Künstliche Intelligenz, Kaiserslautern, Federal Republic of Germany; an acknowledgement of the authors and individual contributors to the work; all applicable portions of this copyright notice. Copying, reproducing, or republishing for any other purpose shall require a licence with payment of fee to Deutsches Forschungszentrum für Künstliche Intelligenz.
ISSN 0946-008X

PPP-Persona: Ein objektorientierter Multimedia-Präsentationsagent

E.André, J. Müller, T.Rist

März 1996

Inhaltsverzeichnis

1	Einleitung	2
2	Anwendungsbeispiele für PPP-Persona	3
2.1	Sprachliche Annotation von Grafiken	3
2.2	Beziehungen zwischen unterschiedlichen Medien	3
2.3	Interaktion des Benutzers mit dem System	4
3	Konzeption des PPP-Persona-Servers	5
3.1	Klassifikation von Persona-Aktionen	5
3.2	Die Architektur des Persona-Servers	7
3.3	X11-Implementation des Persona-Servers	9
4	Planung von Persona-Aktionen in PPP	10
5	Verwandte Arbeiten	12
6	Ausblick	12

Zusammenfassung

In zukünftigen Benutzerschnittstellen werden zunehmend animierte Präsentationsagenten eingesetzt. Der am DFKI entwickelte Präsentationsagent PPP-Persona dient dazu, den Benutzer durch eine Hypermedia-Präsentation zu führen. Dank seiner objektorientierten Konzeption ist er jedoch auch für eine Vielzahl weiterer Anwendungen einsetzbar, wie beispielsweise Online-Hilfesysteme, Home-Shopping und Tele-Banking. Dem Paradigma der Client/Server-Architektur folgend, empfängt der PPP-Persona-Server Präsentationsaufgaben von einem Anwendungsprogramm, die dieser dann selbständig ausführt. Zur visuellen Realisierung des Präsentationsagenten kann sowohl auf Cartoon-Zeichnungen als auch auf Videobildmaterial zurückgegriffen werden. Um die Präsentation lebendiger zu gestalten, kann PPP-Persona selbständig Animationssequenzen initiieren.

1 Einleitung

Window-basierte Benutzungsoberflächen sind bei modernen Computer-Systemen weit verbreitet. Multimedia-Präsentationen umfassen geschriebenen und gesprochenen Text, Soundeffekte, 2D-, 3D-Grafik sowie Animationen. Der nächste Schritt in der Entwicklung von Benutzerschnittstellen ist die personalisierte Benutzerschnittstelle. Einerseits bezieht sich Personalifizierung auf die Möglichkeit des Systems, sich jederzeit auf die individuellen Anforderungen eines Benutzers einstellen zu können. Dies umfaßt die Auswahl der zu präsentierenden Daten und den Präsentationsstil. Dieser Aspekt wird im Bereich der Benutzermodellierung auf breiter Basis untersucht. Andererseits soll sich das System selbst dem Benutzer personalisiert präsentieren, beispielsweise durch eine animierte Figur, die PPP-Persona (Abbildung 1).

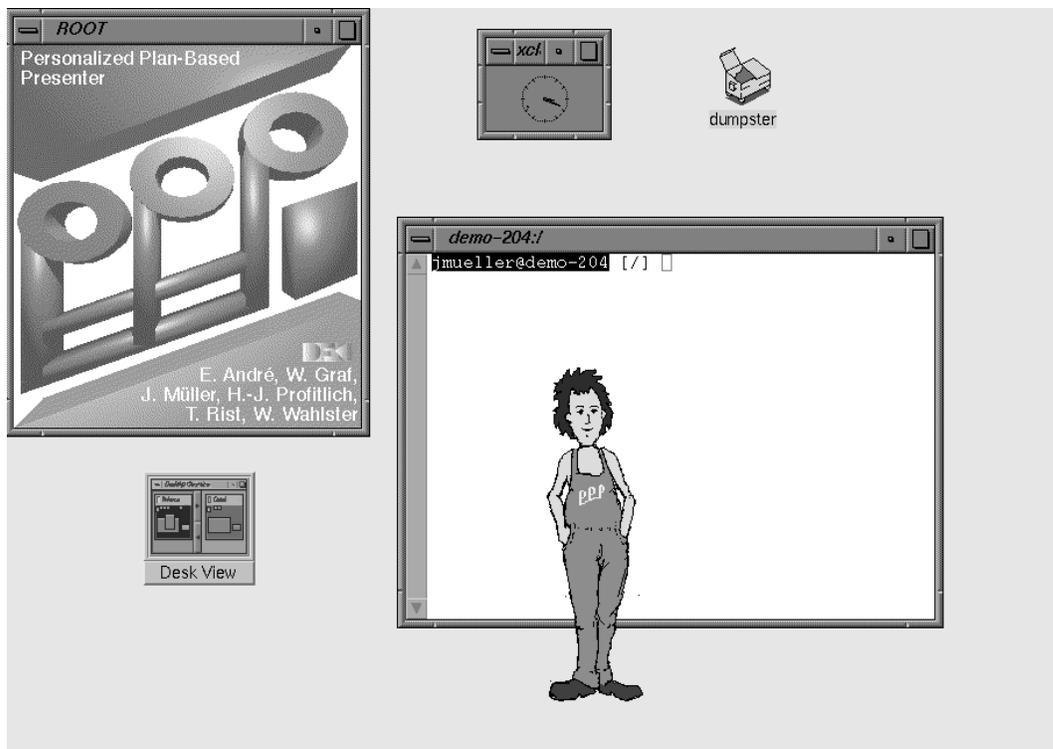


Abbildung 1: PPP-Persona

Ein Präsentationssystem erhält durch einen animierten Präsentationsagenten eine deutlich höhere Ausdrucksstärke, da der Blick des Betrachters durch Zeigegesten des Präsentationsagenten auf wichtige Bereiche gelenkt werden kann. Durch zweihändige Zeigegesten können Beziehungen zwischen zwei Bildschirmobjekten (z.B. Windows, Texte, Grafiken) unterstrichen werden. Durch die per-

sonifizierte Darstellung des Systems lassen sich Systemzustände durch Mimik des Präsentationsagenten ausdrücken. Beispielsweise kann der Präsentationsagent ein verärgertes Gesicht aufsetzen, wenn der Benutzer ein falsches Paßwort eingetippt hat. Der Präsentationsagent hebt sich durch seine äußere Erscheinung von anderen Bildschirmobjekten ab und kann daher zur einfacheren Orientierung des Benutzers bei Präsentationen mit vielen Windows dienen. Der Unterhaltungswert einer Präsentation kann durch Verwendung eines animierten Präsentationsagenten stark gesteigert werden und kann dazu beitragen, die Hemmschwelle bei Computer-Anfängern abzubauen.

2 Anwendungenbeispiele für PPP-Persona

Der vorgestellte Präsentationsagent ist Teil des am DFKI entwickelten Intellimedia-Systems PPP. PPP ist das Nachfolgesystem des wissensbasierten Präsentationssystems WIP. WIP plant und generiert Texte und Grafiken, ausgehend von einer Wissensbasis, die z.B. die Drahtrahmenmodelle der zu präsentierenden Objekte enthält, unter Berücksichtigung verschiedener Präsentationsparameter (Sprache, Benutzervorwissen, Dokumenttyp usw.) . Das Nachfolgesystem PPP muß, da es im Gegensatz zu WIP Online-Präsentation erzeugt, zusätzlich die zeitliche Koordination der einzelnen Präsentationsschritte berücksichtigen.

2.1 Sprachliche Annotation von Grafiken

Eine Beispieldomäne des Systems PPP ist die Generierung und Präsentation von technischen Bedienungsanleitungen. Abbildung 2 zeigt PPP-Persona bei der Präsentation einer Platine eines Modems. Zuerst hat das System ein Window mit der Grafik generiert. Nach dem Aufbau der Grafik muß sich PPP-Persona einen geeigneten Platz in der Nähe des Windows suchen und kann nun mit der Beschreibung der einzelnen Komponenten beginnen. Bei gedruckten Dokumentationen werden die einzelnen Komponenten oft durch Pfeile und Text-Strings annotiert (Das PPP-System bietet diese Variante ebenfalls an.) . Durch PPP-Persona können die einzelnen Komponenten darüberhinaus durch Zeigegesten und durch Sprachausgabe (mittels eines Sprach-Synthesizers) beschrieben werden. Bei dieser dynamischen Annotationsart kann das System die Geschwindigkeit, mit der die einzelnen Objekte eingeführt werden, bestimmen.

2.2 Beziehungen zwischen unterschiedlichen Medien

PPP-Persona kann auch dazu verwendet werden, Beziehungen zwischen zwei unterschiedlichen Windows herzustellen. In Abbildung 3 führt PPP-Persona dazu eine beidhändige Zeigegeste aus, wobei sie mit einem Zeigestock auf eine Modemkomponente und mit dem anderen Zeigestock auf die dazugehörige Be-

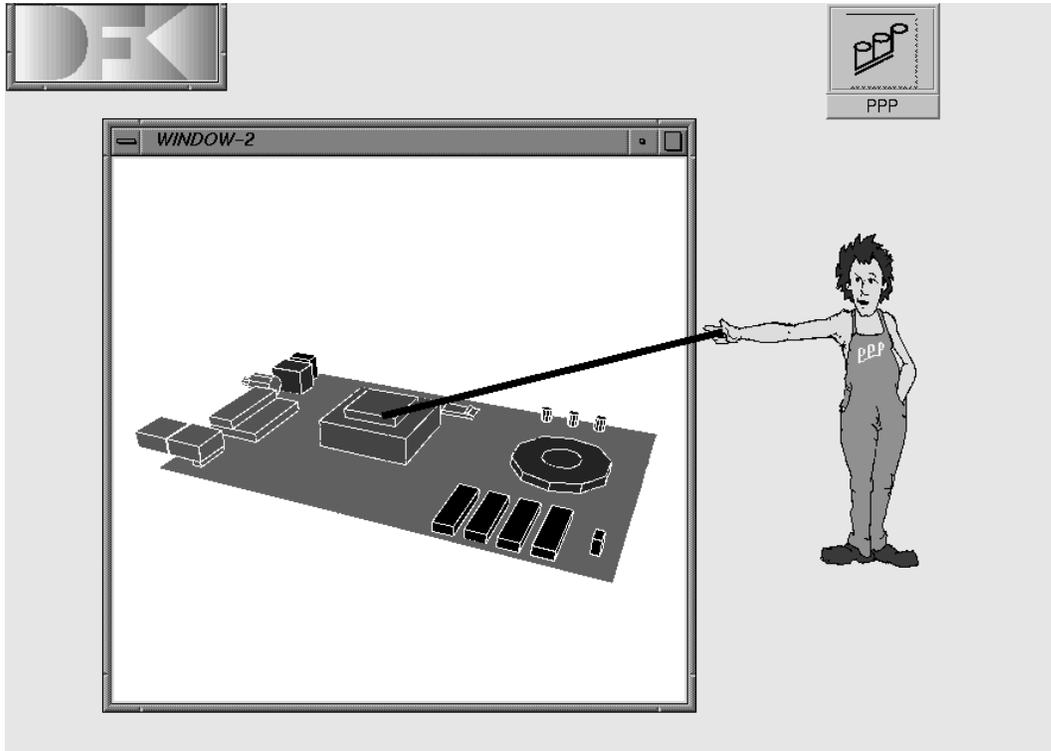


Abbildung 2: Sprachliche Annotation einer Grafik

schreibung zeigt. Gleichzeitig sieht man, daß sich nicht nur gezeichnete Bitmaps zur Visualisierung der Persona einsetzen lassen, sondern daß auch mit Hilfe eines Frame-Grabbers gewonnenenes Video-Material verwendet werden kann.

2.3 Interaktion des Benutzers mit dem System

PPP-Persona dient in der Rolle des personalisierten Präsentationssystems auch als Ansprechpartner für den Benutzer. In PPP wird diese Möglichkeit dadurch ausgenutzt, daß beim Anklicken der Persona das PPP-System-Menü erscheint, so daß der Benutzer dem System Anweisungen geben kann (Abbildung 4).

Werden vom System mehrere Parameter benötigt, so werden diese oft zusammen in einem Fenster vom Benutzer abgefragt. Eine andere Möglichkeit ist, jeden Parameter einzeln unter Kontrolle von PPP-Persona abzufragen, wobei diese zu jedem Parameter Erläuterungen geben kann. Beispielsweise wartet in Abbildung 5 PPP-Persona auf die Eingabe eines numerischen Wertes.

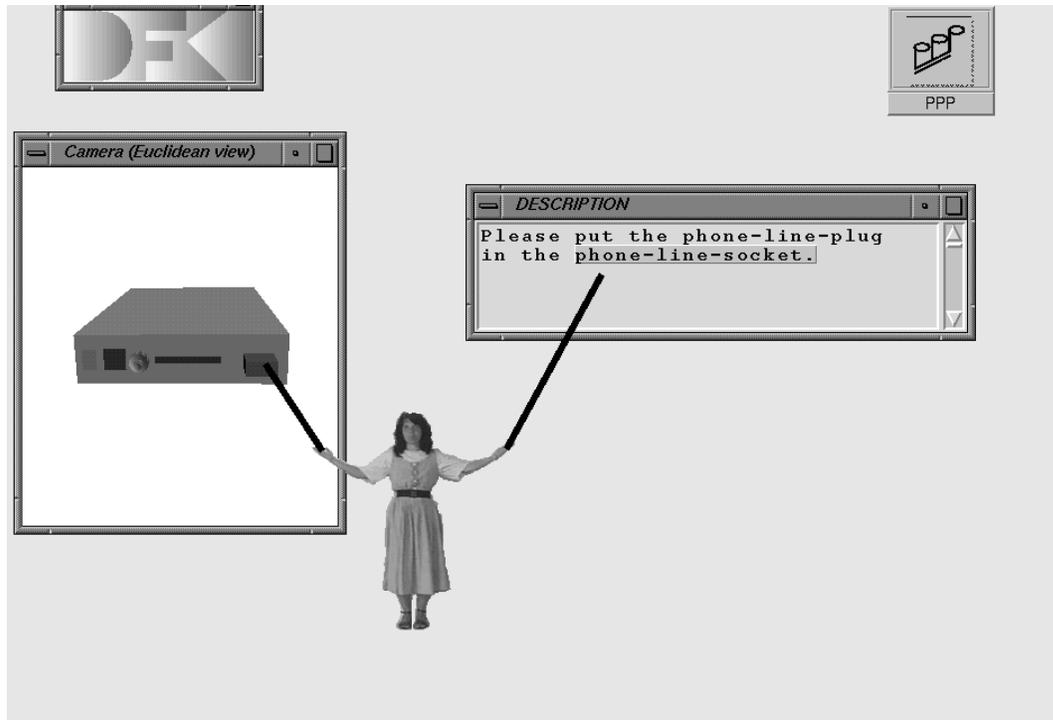


Abbildung 3: Beidhändiges Zeigen mit einer realen Persona

3 Konzeption des PPP-Persona-Servers

Die vom Persona-Server zur Verfügung gestellten Aktionstypen sollten möglichst unabhängig von der jeweiligen Anwendung sein. Es müssen eine große Anzahl von Aktionstypen verfügbar sein, damit Persona möglichst universell einsetzbar ist. Persona soll ein möglichst lebendiges Erscheinungsbild aufweisen. Schließlich soll der Persona-Server weitgehend von der Applikation und von der verwendeten Hardware-Plattform unabhängig sein.

3.1 Klassifikation von Persona-Aktionen

- **Komplexe-Präsentationsakte:** PPP-Persona kann vom jeweiligen Anwendungsprogramm (z.B. dem Präsentationsplaner des PPP-Systems) Aufgaben erhalten. Diese Aufgaben können z.B. Zeigeoperationen, Bewegungsakte, Sprechakte, oder Emotionsäußerungen (sich ärgern, lachen usw.) sein.
- **Ruhe-Aktionen:** Damit Persona möglichst lebendig auf den Betrachter wirkt, muß sie sich auch in Phasen, in denen keine Präsentationsakte ausgeführt werden müssen, bewegen. Im Gegensatz zu den vom Anwendungs-

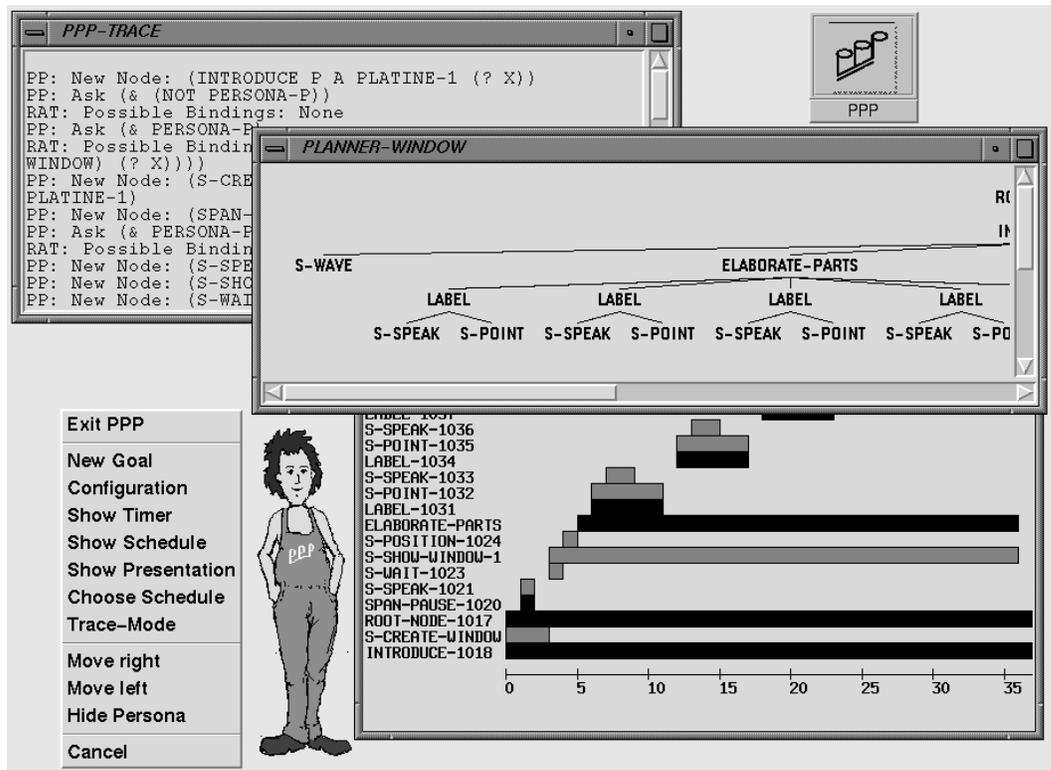


Abbildung 4: PPP-Persona zeigt das PPP-Systemmenü

programm initiierten Aktionen, muß Persona solche Aktionen selbständig starten.

- **Reaktive Aktionen:** Persona muß auf Benutzerinteraktionen umgehend reagieren können. Beispielsweise muß, falls der Benutzer im PPP-System PPP-Persona mit der Maus anklickt, ein Menü erzeugt werden, mit dem der Benutzer das System steuern kann. Auch muß sich Persona umgehend an eine andere Position begeben, falls ihr vom Benutzer mitgeteilt wird, daß sie einen Bildschirmbereich verdeckt, den der Benutzer gerade betrachten will.
- **Primitive Navigationsaktionen:** Primitive Navigationsaktionen umfassen z.B. „Laufen zu einer gegebenen Position“, „Springen“, „Nach-links-drehen“ oder „Nach-rechts-drehen“.
- **Elementaraktionen:** Alle Aktionen, die sich nicht weiter in primitivere Aktionen aufteilen lassen, sind Elementaraktionen. Elementaraktionen bestehen entweder aus einzelnen Bildern oder aus nicht unterbrechbaren Bildsequenzen.



Abbildung 5: Persona wartet auf Benutzereingabe

Die Zerlegung von komplexen Aktionen in primitive Aktionen führt oftmals zu einer Spezialisierung (in Bezug auf den jeweiligen Kontext). Beispielsweise kann die komplexe Aktion Point-To-Window je nach der Lage des Fensters in Bezug zu Persona zerlegt werden in R-Stick-Pointing oder in L-Stick-Pointing (Abbildung 6). Jeder Aktionsauftrag für den Persona-Server muß mit einer Zeitangabe versehen werden, die als Grundlage für die zeitliche Koordination der einzelnen Schritte dient.

3.2 Die Architektur des Persona-Servers

Der Persona-Server stellt eine Schnittstelle zum jeweiligen Anwendungsprogramm und eine Schnittstelle zum Betriebssystem zur Verfügung (Abbildung 7). Der Persona-Server erhält seine Aufträge vom jeweiligen Anwendungsprogramm über das Applikations-Interface. Jedem Auftrag wird vom Persona-Server eine eindeutige Auftragsnummer zugewiesen. Über diese Auftragsnummer kann das Anwendungsprogramm den Status des Auftrages ermitteln oder auch einen Auftrag stornieren. Im Fall des PPP-Systems erhält der Persona-Server seine Aufträge vom PPP-Präsentationsplaner [2] und von einem Constraint-basierten Window-Management-System [9]. Außerdem gibt es eine in LISP-implementierte Testum-

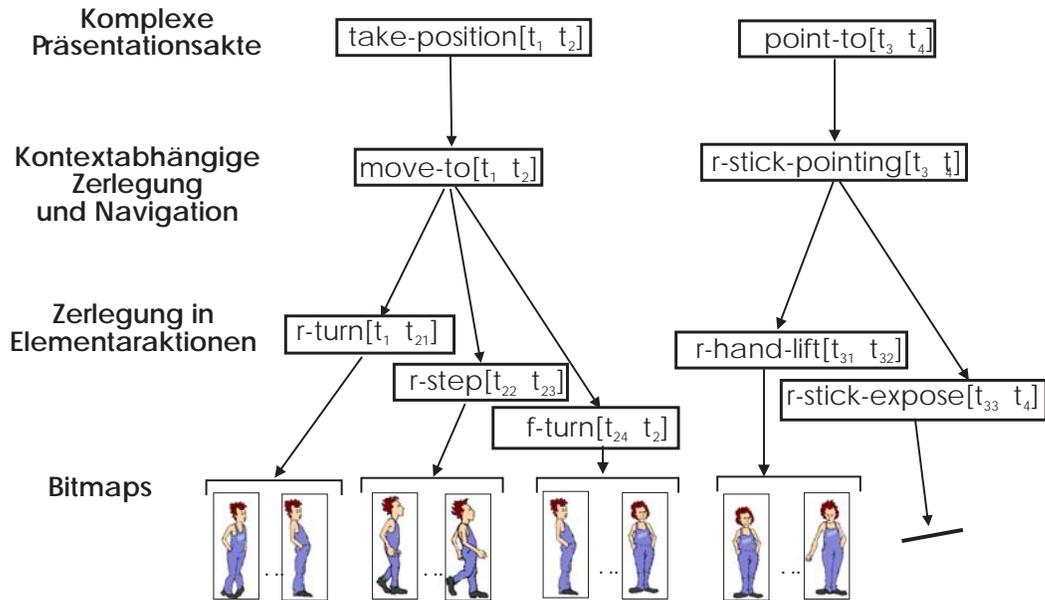


Abbildung 6: Generierung von Persona-Gesten

gebung, so daß Persona-Aufträge in LISP-Syntax eingegeben werden können. Alle Aufträge werden mittels RPC (remote procedure call) an den Persona-Server weitergeleitet.

Der Persona-Server kommuniziert über sein Plattform-Interface mit dem Betriebssystem, dem Window-System und externen Programmen wie Sprach-Synthesizer und Audioplayer. Umgekehrt werden Events (z.B. Mausklicks, Refresh-Events) dem Persona-Server über das Plattforminterface übermittelt. Die internen Komponenten des Persona-Servers umfassen einen Verhaltensmonitor, einen Event-Handler und einen Gesten-Konstruktor.

Der Event-Handler empfängt über das Plattform-Interface Events und prüft, in welcher Weise Persona auf diesen Event reagieren muß. Falls ein Event empfangen wird, für den eine besondere Reaktion von Persona definiert ist (z.B. Refresh-Event, Mausklick des Benutzers auf Persona), wird der entsprechende Auftrag an den Verhaltensmonitor erteilt.

Der Gesten-Konstruktor hat die Aufgabe, aus der Gesten-Datenbasis, die gezeichnete Bilder oder Videobilder enthält, passende Bitmaps auszuwählen und sie zu einer Bildsequenz zusammenzustellen. Er muß außerdem gegebenenfalls das Zeigewerkzeug der Bitmap hinzufügen.

Der Verhaltensmonitor ist die zentrale Komponente des Persona-Servers. Der Verhaltensmonitor wartet auf eine Eingabe vom Applikationsinterface oder vom Event-Handler. Falls ein Auftrag vom Anwendungsprogramm vorliegt, wird dieser

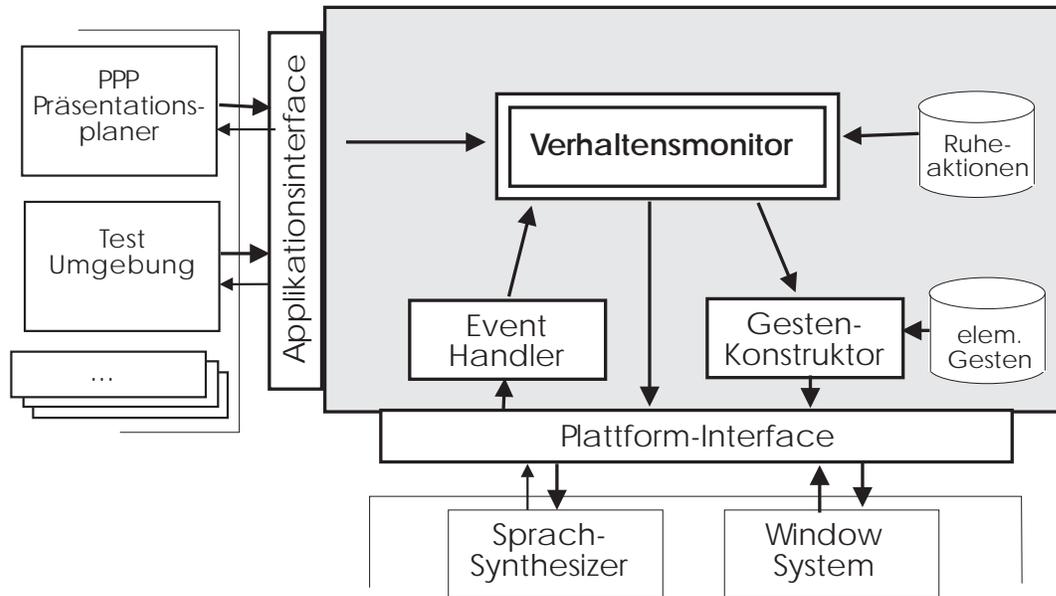


Abbildung 7: Aufbau des Persona-Servers

komplexe Auftrag in primitive Aktionen (in Abhängigkeit vom jeweiligen Kontext) zerlegt und auf einem Stack abgelegt. Gleichzeitig wird dem Event-Handler ein Event gesendet, daß primitive Aktionen existieren. Der Event-Handler sorgt in Kooperation mit dem Verhaltensmonitor und dem Gesten-Konstruktor für die Abarbeitung der auf dem Stack gespeicherten Aktionen. Falls der Benutzer mittels der eindeutigen Aktionsnummer eine Aktion abbricht, sorgt der Verhaltensmonitor für ein ordnungsgemäßes Beenden der Aktion. Stellt der Event-Handler fest, daß momentan keine Aktionen auszuführen sind, d.h. ist der Stack leer, erteilt er dem Verhaltensmonitor den Auftrag, eine Ruhe-Aktion auszuwählen. Der Verhaltensmonitor speichert die dekomponierte Ruheaktion auf dem Stack, welche dann analog zu einem von einem Anwendungsprogramm initiierten Auftrag abgearbeitet wird.

3.3 X11-Implementation des Persona-Servers

Die Implementation des Persona-Servers basiert auf X11 unter Verwendung der X11-Shape-Extension. Mit Hilfe der X11-Shape-Extension ist es möglich, unregelmäßige Windows zu definieren. Dies ist notwendig, da das Persona-Window nur die Größe der Figur haben soll. Dadurch werden Events (Maus-Events, Tastatur-Events) nur dann an den Persona-Server weitergeleitet, wenn der Mauszeiger sich innerhalb der Persona-Bitmap befindet. Die Fläche des Windows wird mit Hilfe einer Bitmaske definiert (Abbildung 8).

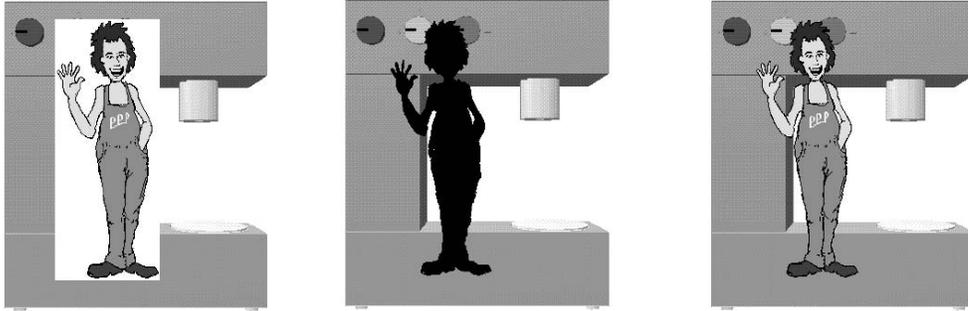


Abbildung 8: Vergleich rechteckiges Window und unregelmäßiges Window

Diese Bitmaske setzt sich aus der Bitmaske der Persona-Bitmap und aus der Bitmaske der Zeigewerkzeuge zusammen. Der Persona-Server berechnet die Bitmasken für die Persona-Gesten in der Startphase und speichert sie zusammen mit den Persona-Bitmaps im Speicher des X-Servers. Die Bitmasken bzw. die Bitmaps für die Zeigewerkzeuge werden nur auf Anfrage zur Laufzeit berechnet. Zur Ausgabe von Sprache wird ein Sprach-Synthesizer verwendet, der ebenso wie der Audio-Player über den Systemaufruf *exec* aktiviert wird. Falls der Audio-Prozess mehr Zeit benötigt, als ihm zusteht, wird er mittels *kill* abgebrochen.

Es existieren momentan Implementationen des Persona-Servers für Sun Sparc (Solaris) und für Silicon Graphics (Irix) Workstations (Abbildung 9).

4 Planung von Persona-Aktionen in PPP

Die Generierung von Multimedia-Präsentationen wird in dem System PPP als Planungsproblem aufgefaßt. Ausgehend von einem kommunikativen Ziel erzeugt ein Präsentationsplaner [2] einen Präsentationsplan, dessen primitive Aktionen an die einzelnen Generatoren, wie Grafikgenerator, Textgenerator usw., weitergereicht werden. Ein Beispiel für ein kommunikatives Ziel ist z.B. die Aufgabe, dem Benutzer eine Modemplatine zu erläutern: (Introduce System User circuit-board-1 ?window). Wird der Präsentationsplaner auf ein solches Ziel angesetzt, sucht er nach einer geeigneten Präsentationsstrategie. Im Beispiel wäre folgende Strategie anwendbar:

(S1) **Header:** (Introduce System User ?object ?window)

Inferiors:

((A1 (Create-Graphics P A ?object ?window))

(A2 (S-Show-Window System User ?window ?object))

(A3 (S-Wait System User))

(A4 (S-Position Persona User))

(A5 (Elaborate-Parts Persona User ?object ?window))

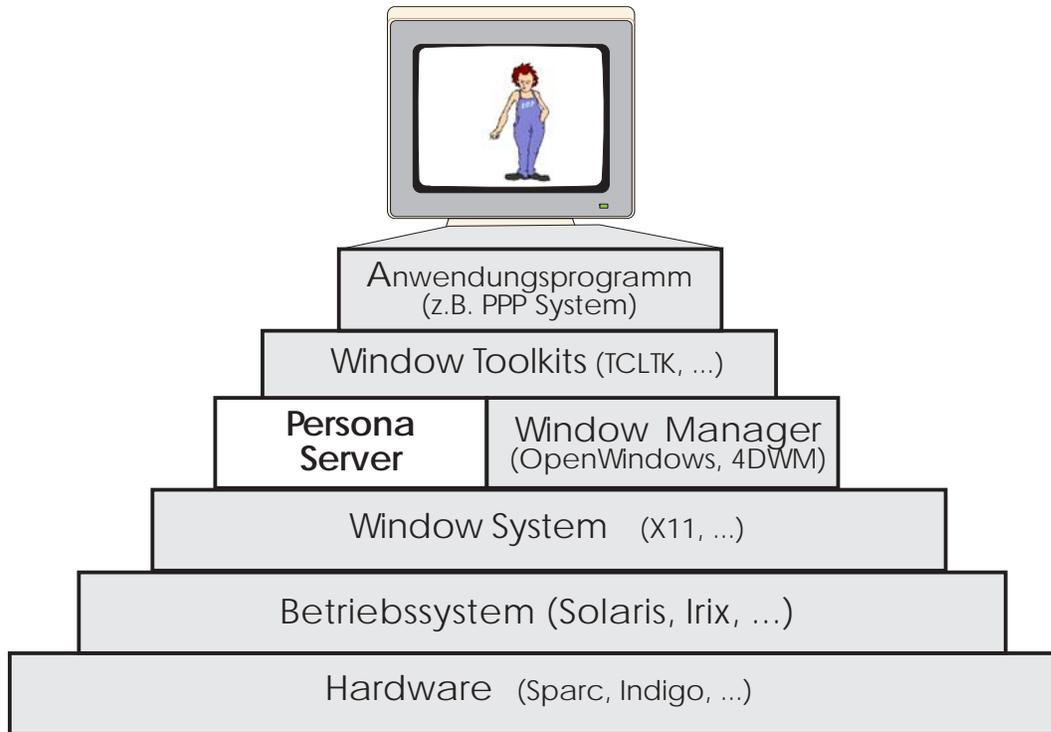


Abbildung 9: Integration des Persona-Servers in das Gesamtsystem

Qualitative:

((A1 (before) A2) (A3 (starts) A2) (A3 (meets) A4) (A2 (equals) A5))

Metric:

((10 ≤ Duration A2 ≤ 40) (1 ≤ Duration A4 ≤ 2))

Start: A1

Entsprechend dieser Strategie wird zunächst eine Grafik für die Modemplatine generiert. Dann wird ein Window geöffnet, und nach einer kurzen Pause begibt sich Persona in eine geeignete Zeige-Position. Anschließend werden die Platinenkomponenten dem Betrachter erläutert. Während S-Show-Window, S-Wait, S-Position aus der Sicht des Planers primitive Aktionen sind, die unmittelbar an die Generatoren (Window-Management-System bzw. Persona-Server) weitergeleitet werden, sind Create-Graphics und Elaborate-Parts komplexe Operationen, die vom Planer weiter zerlegt werden. Die weitere Zerlegung von Elaborate-Parts führt zu Zeigegesten und Sprechakten. Die Zeigegesten werden vom Persona-Server ausgeführt. Da das System über eine explizite Repräsentation der dargestellten Objekte verfügt, kann es die Position der abgebildeten Objekte einfach bestimmen und zur Generierung von Zeigegesten an den Persona-Server senden.

Der vom Sprach-Synthesizer auszugebende Text wird mittels eines Textgenerators [12] automatisch erzeugt. Analog führt die Zerlegung des Teilzieles Create-Graphics zu einem Aufruf an den Grafikgenerator [17, 14], der eine adäquate Abbildung der Modemplatte erstellt. Die zeitliche Koordinierung der einzelnen Handlungsschritte wird durch zeitliche Constraints festgelegt [11].

5 Verwandte Arbeiten

Animierte Interface-Agenten und Präsentatoren wurden auch von anderen Autoren vorgeschlagen [13, 7]. Diese sind allerdings meistens auf besondere Anwendungsgebiete spezialisiert. Das Ziel bei der Implementierung des PPP-Persona-Servers ist hingegen die Entwicklung eines anwendungsunabhängigen animierten Präsentationsagenten. Dabei gibt es Berührungspunkte zu folgenden anderen Projekten:

In [8] wird die Ausgabe des X-Servers mit Videoclips von realen Personen gemischt, so daß reale Personen in Computergrafiken als Präsentator auftauchen können. Zu diesem Zweck wird spezielle Video-Hardware verwendet. Bei der Implementierung des PPP-Persona-Servers wird dagegen der Standard-X-Server verwendet, aus dessen Sicht PPP-Persona ein normales Window ist. Der PPP-Persona-Server setzt keine spezielle Video-Hardware voraus.

Der in [5] beschriebene Animations-Server dient als Hilfesystem bei der Benutzung von Window-Oberflächen (z.B. Menü öffnen usw.) Er folgt ebenso wie der PPP-Persona-Server der Client/Server-Philosophie, d.h. ein Anwendungsprogramm sendet Animations-Aufträge an den Animations-Server, der diese Aufträge bearbeitet.

Ein sich in der Entwicklung befindendes System verwendet einen dreidimensional modellierten Papagei, der den Benutzer bei der Wahl von Audio-CD's unterstützt [4]. Allerdings beschränkt sich dieses System auf das Abspielen von Multimediamaterial.

Auch die Verwendung von 3D-Gesichts-Animation [10, 16] kann bei der Implementierung von lebendig wirkenden Präsentationsagenten zur Modellierung der Gesichtsmimik Verwendung finden.

6 Ausblick

Der PPP-Persona-Server ist ein Präsentationsagent, der durch seine vielfältigen Möglichkeiten die Ausdruckstärke einer grafischen Benutzungsoberfläche erweitern kann. Ein Anwendungsprogramm kann dem Persona-Server auf einem hohen Abstraktionsniveau Befehle erteilen, welche er dann selbständig ausführt. Durch Verwendung von Ruhe-Aktionen erhält PPP-Persona ein natürlicheres und lebendigeres Erscheinungsbild. Der Benutzer kann unmittelbar mit PPP-Persona

kommunizieren und dem durch PPP-Persona personifizierten Anwendungsprogramm Aufträge erteilen.

Obwohl PPP-Persona in dem PPP-System ein erstes Anwendungsbeispiel gefunden hat, ist er doch dank seiner portablen Realisierung in einer Vielzahl von anderen Anwendungsgebiete einsetzbar. In Applikationen aus den Bereichen Tele-Shopping oder Home-Banking könnte Persona als Verkäufer bzw. als Berater auftreten. In Telekonferenzen könnte Persona (durch Verwendung von digitalisiertem Videomaterial) die Rolle eines der Konferenzteilnehmer übernehmen.

Literatur

- [1] **B.Adelson**. *Evocative Agents and Multi-Media Interface Design*. In: UIST, S. 351-356, 1992.
- [2] **E.André, T.Rist**. *Towards a Plan-Based Synthesis of Illustrated Documents*. In: Proc. of the 9th ECAI, S.25-30, Stockholm, 1990.
- [3] **E.André, T.Rist**. *Generating Coherent Presentations Employing Textual and Visual Material*. Artificial Intelligence Review, Special Volume on the Integration of Natural Language and Vision Processing,9(2-3):147-165,1995.
- [4] **G.Ball, D. Ling, D.Kurlander [u.a.]**. *Lifelike Computer Characters: the Persona project at Microsoft Research*. Microsoft, 1995.
- [5] **K. Bharat, P.N. Sukaviriya**. *Animating User Interfaces Using Animation Servers*. UIST, S.69-79,1993.
- [6] **M.F.Costabile, T.Catarci, S.Leviadi**. *Advanced Visual Interfaces (Proceedings of AVI '92, Rome, Italy)*. Singapore: World Scientific Press, 1992.
- [7] **A.Don, T.Oren, B.Laurel**. *Guides 3.0*. In: CHI-93, Video-Proceedings, 1991.
- [8] **S.Gibbs, C.Breiteneder, V. de Mey, M. Papathomas**. *Video Widgets and Video Actors*. In: UIST, S. 179-184, 1993.
- [9] **W.H.Graf, S.Neurohr**. *Constraint-Based Layout in Visual Program Design*. In: Proceedings of the 11th International IEEE Symposium on Visual Languages (VL '95), Darmstadt, 1995.
- [10] **P.Kalra, A.Mangili, N.Magenat-Thalmann, D.Thalmann**. *Smile: A Multilayered Facial Animation System*. In: T.L. Kuni (Hrsg.), Modeling in Computer Graphics. New York: Springer, 1991.
- [11] **H.A.Kautz, P. B. Ladkin**. *Integrating Metric and Qualitative Temporal Reasoning*. In: Proceedings of AAAI '91, 1991.

- [12] **A.Kilger.** *Using UTAGs for Incremental an Parallel Generation.* Computational Intelligence, 10(4):591-603, 1994.
- [13] **B. Laurel.** *The Art of Human-Computer Interface Design* Addison-Wesley, 1990.
- [14] **J.Müller.** *GeoDisplay: Ein flexibles 3D-Visualisierungswerkzeug für Lisp.* Diplomarbeit an der Universität des Saarlandes. Saarbrücken: 1995.
- [15] **K.Packard.** *X11 Nonrectangular Window Shape Extension Version 1.0, X11 R5.* Technical report, MIT X Consortium, MIT, Cambridge, Massachusetts, 1989.
- [16] **C.Pelachaud.** *Functional Decomposition of Facial Expressions for an Animation System.* In: [6].
- [17] **T.Rist, E.André.** *Incorporating Graphics Design and Realisation into the Multimodal Presentation System WIP.* In: [6].
- [18] **W.Wahlster, E.André, W.Finkler, H.-J.Profitlich,T.Rist.** *Plan-Based Integration of Natural Language an Graphics Generation.* AI Journal, 63:387-427, 1993.